

# For Friday

- Chapter 9, sections 2-3
- Program 3, part 1 due

# Research Paper

- Any questions?

# Program 3

- Any questions?

# Path Compression

- Try to improve the height of our trees while changing time cost by a constant factor

# Maze Generation

# Graphs

- A **graph** is a collection of **vertices** (singular **vertex**) and **edges**
- Vertices are sometimes called **nodes**
- Vertices are usually named, edges are identified by the vertices they connect: **AB** is an edge connecting vertex **A** to vertex **B**

# Graphs (cont)

- We will denote the number of vertices in a graph by  $V$  and the number of edges by  $E$
- A **path** from vertex  $A$  to vertex  $B$  is a listing of the vertices touched as we traverse edges from  $A$  to  $B$ .
- If there is a path from every node to every other node in a graph, the graph is **connected**.

# More Graphs

- A **cycle** is a path which begins and ends at the same vertex.
- A **simple path** is a path with no cycles in it.
- A graph with no cycles is called a **tree**.
- A **spanning tree** of a graph is a tree subgraph that contains all the vertices in the original graph.

# Even More Graphs

- A graph with fewer than  $V-1$  edges cannot be connected.
- A graph with  $V$  or more edges must contain a cycle.
- **Complete** graphs have all possible edges present.
- **Dense** graphs have most possible edges present.
- **Sparse** graphs have few of the possible edges present.

# Types of Graphs

- Graphs with edges that can go either way are **undirected**.
- Graphs with edges that have particular direction are **directed**.
- Graphs may also have weights associated with their edges. Those are **weighted** graphs.
- Weighted graphs may be either directed or undirected.

# Graph Representation

- Adjacency matrix
  - Graph is represented using an 2-d array
  - Indices of array represent nodes in the graph
  - Cells of the matrix represent edges
- Adjacency lists

# Graph Applications

- ???

# Graph Applications

- Parallel computing
- Scheduling
- Games
- Problem solving
- Mapquest
- Networks (of wires, pipes, etc.)
- Knowledge representation